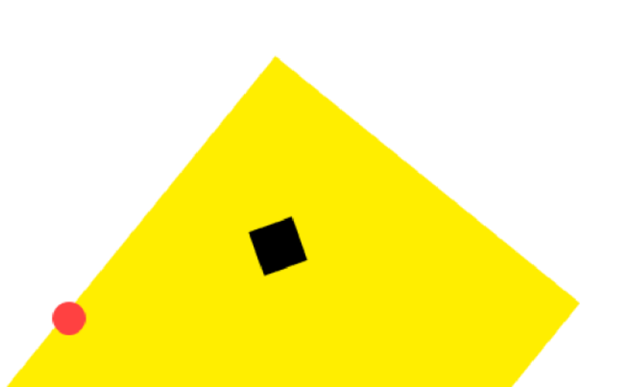
# Webpack: заменить нельзя оставить

Кувшинов Евгений



“Распространённые инструменты сборки в **10-100x** медленнее, чем они могут быть”

**Эван Уоллес**



# Евгений Кувшинов

- Фронтендер
- Тренер по инженерным практикам
- 5 лет развивал фронтенд и инженерную культуру в ManyChat



# Что вы используете для сборки?

**Webpack** **Rollup** **Vite** **swc** **ESBuild**



# Сколько времени занимает сборка?

1+ мин

10+ сек

1-10 сек



# У меня в начале 2021 года:

– Webpack
– 3+ мин



# История повторяется

THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:

"MY CODE'S COMPILING."

HEY! GET BACK TO WORK!

COMPILING!

OH. CARRY ON.



```
kib357@kibmac project % npm start
█
```



# План

- Почему я стал искать альтернативы Webpack?
- Bundler, compiler, dev-tool: кто есть кто?
- Что предлагают SWC и ESBuild
- Замена Webpack на проекте с шестилетней историей разработки



# Где болит от медленной сборки?



# Локальная сборка

## при разработке



# Сборка на CI

## при интеграции



# На что влияет скорость сборки?

Lead time for changes

Deployment frequency

Time to restore

Developer experience



# На что влияет скорость сборки?

## Метрики и успешность организации



# Цели:

**Пайплайн CI за 4 минуты**
сборка+линтер+тесты+деплой

**Улучшить DX**
метрика: фидбэк от разработчиков



# В путь!

## Но сначала немного о процессе сборки…



# Bundler, build tool, transpiler…

## Кто есть кто?



Babel · The compiler for writin ×

Loiane

Secure | https://babeljs.io/repl#?babili=false&browsers=...

BABEL

```
console.log('Babel');
```

```
'use strict';

console.log('Babel');
```



# Source-to-source compiler (transpiler)

Babel · The compiler for writing ×

Loiane

Secure | https://babeljs.io/repl#?babili=false&browsers=...

BABEL

```
console.log('Babel');
```

```
'use strict';

console.log('Babel');
```



webpack

DOCUMENTATION CONTRIBUTE VOTE BLOG

Search ⌘ K

# bundle your scripts

.js .js .hbs .png .sass .cjs .jpg .sass .sass

MODULES WITH DEPENDENCIES

.js .css .jpg .png

STATIC ASSETS



# Bundler

webpack

DOCUMENTATION CONTRIBUTE VOTE BLOG Search ⌘ K

bundle your scripts

.js .hbs .cjs .sass .js .sass .sass .png .jpg

MODULES WITH DEPENDENCIES

.js .css .jpg .png

STATIC ASSETS



# Задачи Webpack

- Построить граф зависимостей
- Загрузить и преобразовать зависимости
- Скомпоновать и сохранить их в новой структуре



# Webpack + Babel = ❤️

webpack

DOCUMENTATION CONTRIBUTE VOTE BLOG

Search ⌘ K

bundle your scripts

.js
.js
.hbs
.png
.sass
.cjs
.jpg
.sass
.sass

BABEL

.js .css
.jpg .png

STATIC ASSETS

MODULES WITH DEPENDENCIES



Vite

Next Generation
Frontend Tooling

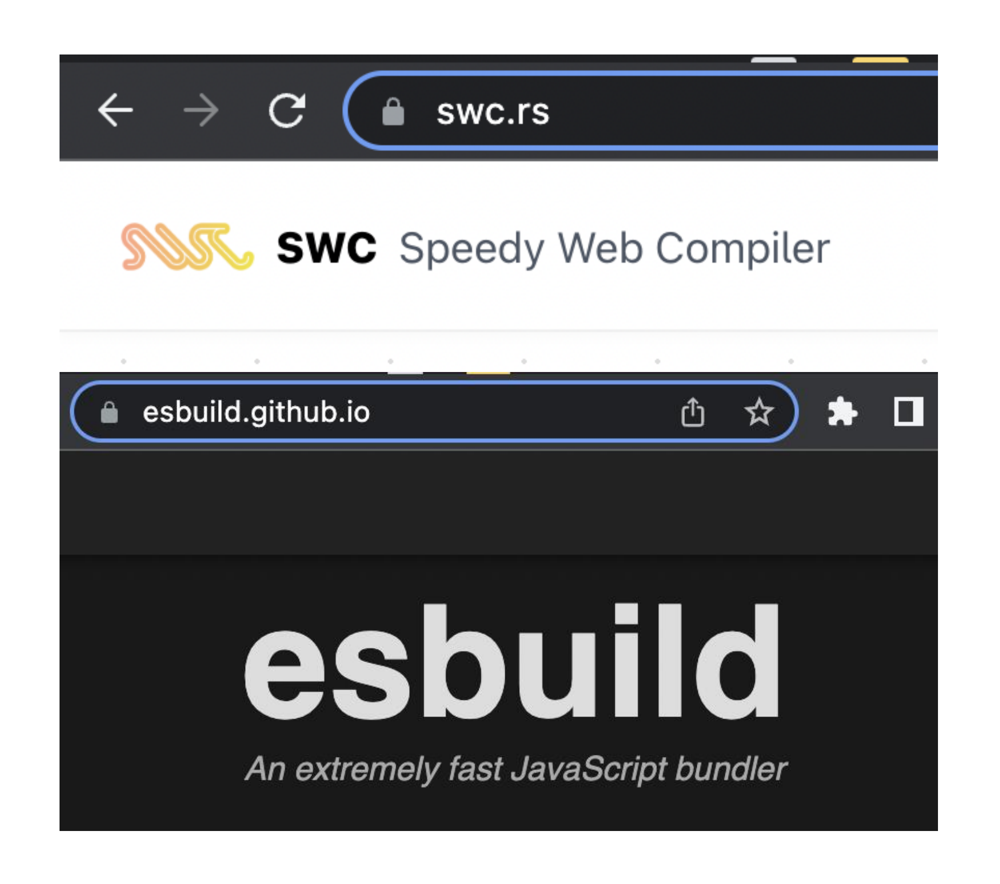
Get ready for a development environment that can finally catch up with you.

Get Started | Why Vite? | View on GitHub

**“Vite is a build tool… It consists of two major parts: dev server and build command that bundles your code with Rollup”**



# А что с ESBuild и SWC?




"SWC can be used for both compilation and bundling."

"By default esbuild will ***not*** bundle the input files. Bundling must be explicitly enabled."



# В SWC и ESBuild работает из коробки:

Современный ECMAScript

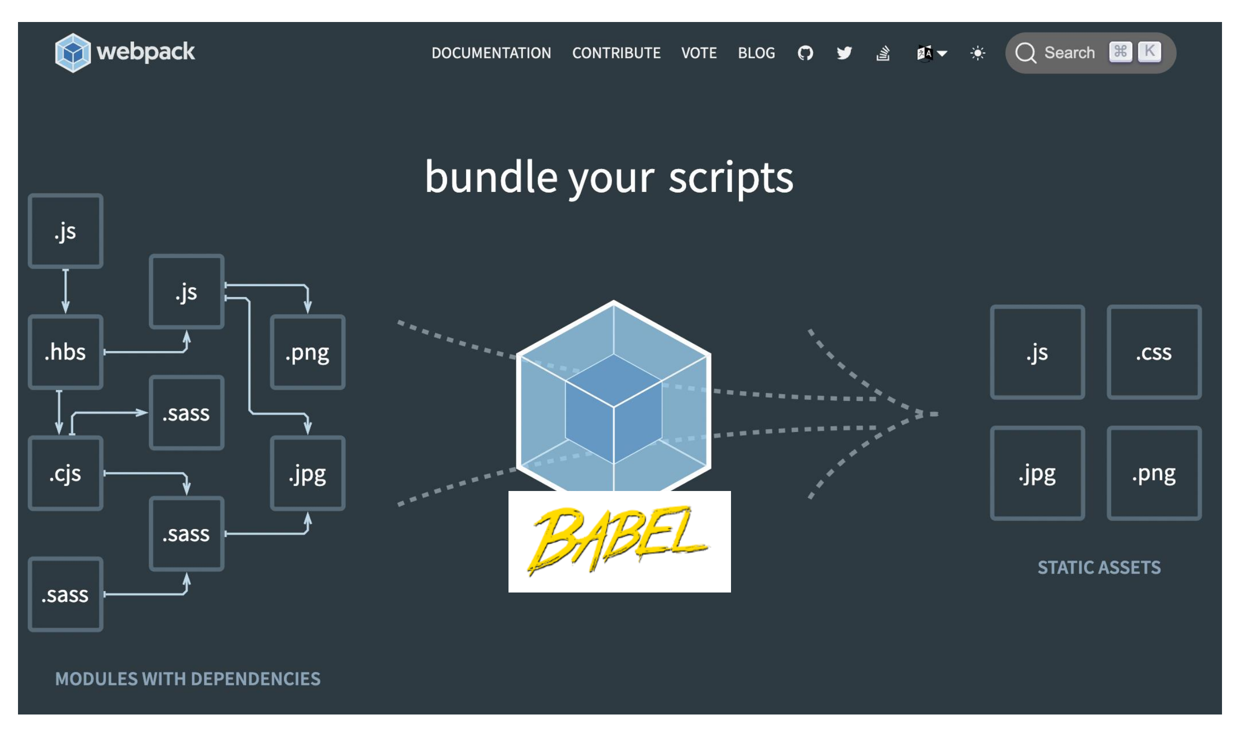
TypeScript

JSX

Minification



# Просто заменим лоадер!









# Лайфхак: новый транспайлер для Webpack может дать 10-30% ускорения!



# Но где обещанные 10-100x?



# Для максимального ускорения нужно заменить бандлер полностью

- Построение графа зависимостей занимает время
- Нужна параллелизация
- Нужно минимизировать переходы Native-JS
- Минификация тоже может быть быстрее



# Может быть, всё-таки ускорить Webpack?



# Мы попробовали:

- Рецепты ускорения через конфигурацию
- Обновления webpack и лоадеров
- Ленивую сборку (experiments.lazyCompilation)
- Кэш



**Нужно заменить Webpack с десятком плагинов и множеством настроек на новый бандлер.**

## Что может пойти не так?



# От этого нужно отказаться:

- Специфические для Webpack импорты

```
//...
import(/* webpackPreload: true */ 'ChartingLibrary');
```

```
require.context('../', true, /\.stories\.js$/);
```

```
import { myFunction } from "exports-loader?exports=myFunction!./file.js";
```



# От этого нужно отказаться:

- Специфические Babel-плагины

```
<If condition={ true }>
  <span>IfBlock</span>
</If>
```



# От этого нужно отказаться:

- Глобальные переменные Webpack

__webpack_public_path__ (webpack-specific)



# План перехода

✅ Подготовка (отказ от специфики Webpack + Babel)
- Выбор бандлера: SWC или ESBuild
- Настройка сборки и замеры: скорость, lighthouse
- Перевод локальной сборки на новый бандлер
- Сбор фидбэка от разработчиков
- Перевод продакшн-сборки на новый бандлер



# SWC или ESBuild?

| | | Stars | Issues | Version | Updated | Created |
|---|---|---|---|---|---|---|
| | @swc/core | 23,811 | 298 | 1.3.0 | 2 days ago | 4 years ago |
| | esbuild | 33,141 | 309 | 0.15.7 | 8 days ago | 5 years ago |



# 3 шага к быстрой сборке



# #1 Настройка ESBuild

- Первоначальная настройка заняла неделю
  - Stylus + CSS modules вызвали максимум сложности

- Сборка:
  - Webpack: 3 минуты
  - ESBuild: **25 секунд**

7.2x



# Но где обещанные 10-100x?



# #2 Отказ от ненужного и устаревшего

- Stylus может занимать больше половины времени сборки

- Сборка без Stylus
    - Webpack: 1.5 минуты
    - ESBuild: **5 секунд**

**18x**



# #3 Кэш

- CSS, бывает, переиспользуется

- Сборка с кэшем CSS
  - Webpack: 1.5 минуты
  - ESBuild: **2.5-3 секунды**

30x



# Неожиданно

- Cypress-тесты на CI стали проходить в **2 раза** медленнее





# Хорошие итоги

- Улучшились метрики в Lighthouse
- Ускорилась сборка в 30 раз
- Ноутбуки разработчиков не шумят вентиляторами, а батареи хватает на дольше



# Итоги не очень

- 3 недели подготовки и 3 недели настройки
- Некоторые плагины придётся писать самостоятельно
- Новички будут приходить без опыта



**Если скорость сборки – не самый важный критерий, можно и**

# оставить Webpack



## Если скорость сборки имеет значение, потребуется

# Webpack заменить





- Поддержка emitDecoratorMetadata
- Поддержка Jest (@swc-jest)

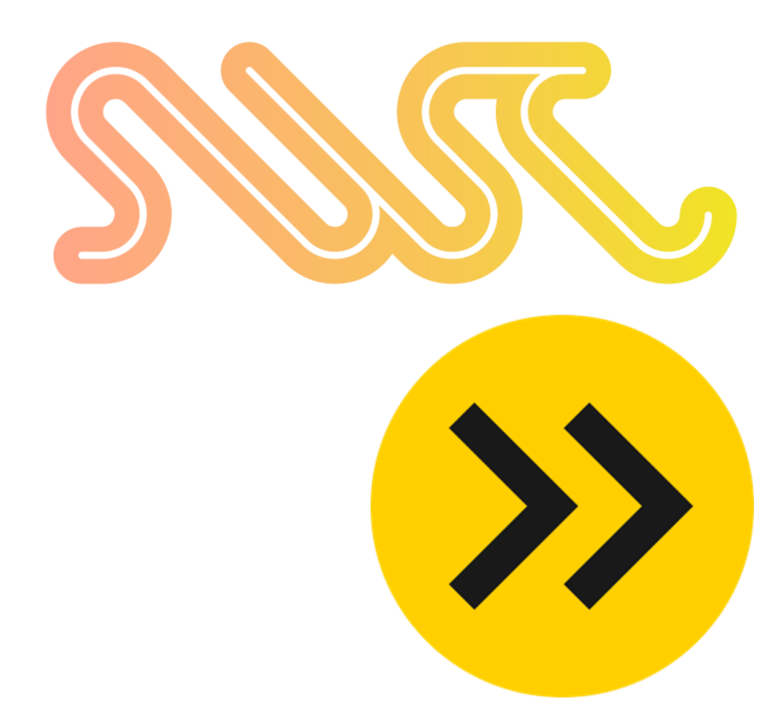


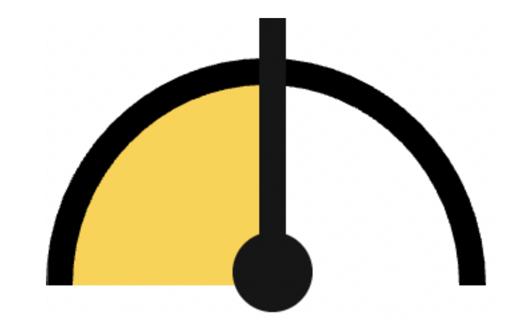
- Простой старт: postcss, glob import и многое другое из коробки
- Абстракция над бандлером














FC
Frontend
Conf 2022

## Спасибо за внимание!

@kib357



